REMARKS

The present reply is responsive to the Office Action dated June 27, 2005. Claims 15, 23, 31, 35, 36 and 37 have been amended. No new matter has been added by these amendments. Claims 1-37 are again presented for the Examiner's consideration in view of the following comments. A petition for a one-month extension of time is submitted herewith.

As an initial matter, the applicant would like to thank the Examiner for the telephone interview with the undersigned on October 3, 2005 regarding the instant application. As indicated in the interview summary dated October 11, 2005, the telephone interview included suggestions by the Examiner for amending independent claim 31, as well as a discussion of the structure and operation of U.S. Patent No. 5,781,750 ("*Blomgren*") and the teachings of the *Hennessy* reference as compared with the claimed invention. As will be described below, applicant has amended claim 31, as well as claims 15, 23 and 35, in view of the Examiner's suggestions.

Features of *Blomgren* and *Hennessy* have been discussed in the prior amendments of the instant application, and those discussions are incorporated by reference for the sake of brevity. Specific details of the references will be addressed below in view of specific rejections.

The Examiner rejected claims 1-2, 4-5, 9-11, 23, 26-30, and 35-37 under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,781,750 to *Blomgren et al.* ("*Blomgren*"). Applicant respectfully traverses the rejection. Independent claims 23 and 35 have been amended in a manner similar to independent claim 31 to include parallel processing operations, and will be addressed along with claim 31 below. The rejection of independent claims 1, 26, 36 and 37 will now be addressed.

As discussed in the telephone conference, it is applicant's understanding that *Blomgren* performs instruction emulation in a two-pass process. A marked-up version of FIG. 2 of *Blomgren* is included in Appendix A to illustrate the two-pass process. Initially, an instruction (either a CISC or RISC instruction) is fetched with an instruction fetch unit 32, which is sent to instruction decode unit 36. (*See* col. 6, ll. 16-20; *see also* FIG. 2.) "Instruction decode 36 is composed of three sub-blocks, one for decoding CISC instructions, another for decoding RISC instructions, and a third sub-block for decoding extended RISC instructions for emulation mode." (Col. 6, ll. 20-24.) Upon decoding by a respective one of the sub-blocks, the instruction is passed through multiplexer 46 to execution unit 48 for processing. (See col. 6, ll. 53-61; *see also* FIG. 2.) According to *Blomgren*, the execution unit can execute simple CISC and simple RISC instructions, but not more complex instructions from either set, as "only simple instructions are directly supported." (Col. 7, ll. 4-5.) This first pass is illustrated as pass "A" in the marked-up version of FIG. 2.

Results from the execution unit are then passed to a translation look-aside buffer (TLB) 52. "If the translation is not present in the TLB, a miss is signaled which causes emulation mode to be entered... When entry to emulation mode is requested, entry point block 56 generates the proper entry point vector or address in the emulation portion of memory, and loads this address into the instruction pointer 34. Thus the CPU will begin fetching and executing instructions at the specified entry point, where the emulation driver contains a routine to handle the exception, TLB miss, or to emulate the unknown instruction. Instruction decode block 36 can provide the opcode itself and other fields of the instruction to the entry point logic, to allow the entry point to be more fully specified." (Col. 7, ll. 25-46.) In this second pass, illustrated as pass "B" in the

marked-up version of FIG. 2, the emulated instruction is passed from the instruction decode block 36 through the multiplexer 46 to execution unit 48.

Thus, it can be seen that *Blomgren* requires two passes to process a complex instruction. The first attempt (pass "A") passes a RISC or CISC instruction from instruction fetch 32 unit through instruction decode 36 and multiplexer 46 to the execution unit 48. If the instruction is not one of the "simple instructions" that are "directly supported" by the execution unit 48, the instruction execution process (pass "B") begins commences.

In the second pass, TLB 52 triggers an entry into emulation mode via mode control logic 42. The mode control logic 42 sets the emulation mode control bits in the mode register 38 and signals the entry point block 56. (*See* FIG. 2.) It is at this point that the entry point block 56 loads an appropriate emulation address into the instruction pointer 34 so that emulation mode instructions can be loaded into the instruction fetch 32. The emulated instruction passed from instruction fetch 32 unit back through instruction decode 36 and multiplexer 46 to the execution unit 48. Such a two pass process is time and resource inefficient.

In contrast, the present invention operates quickly and efficiently in a single pass. By way of example only, a marked-up version of FIG. 2 of the instant application is provided in Appendix A. Here, instructions and other information are passed via different routes from the memory 51 to the instruction selector 108. One route (route "1") passes an instruction (e.g., a computer instruction) from memory 51 to the instruction selector 108. Another route (route "2") passes the instruction such as the computer instruction to the address generator 102, which in turn generates an address based on the

computer instruction and passes the address to the jump instruction generator 104. The jump instruction generator 104 creates an instruction to jump to the address generated by the address generator 104, and passes the jump instruction to the instruction selector 108. A third route (route "3") passes the instruction, such as the computer instruction, from memory 51 to the complex instruction detector 106, which issues an output indicative of whether the instruction is a member of a set of instructions (e.g., whether it is a "complex" instruction) and passes this indicator to the instruction selector 108.

As illustrated, the data from the three routes, e.g., the computer instruction, the jump instruction, and the indicator from the complex instruction detector are provided to the instruction selector 108, which sends either the original computer instruction or the jump instruction to the instruction cache memory 52 based on the indicator. Thus, it can be seen that multiple routes are taken in a single pass, and when two versions of an instruction are available, the instruction selector 108 chooses between them based on the indicator. In contrast with *Blomgren*, there is no first pass attempt to process an instruction followed by a second pass in which an emulation instruction actually processed.

Features of this embodiment are presented in the claims. The system of claim 1 includes a complex instruction detector and an address generator that each input a computer instruction from a source of instructions. The complex instruction detector produces an output indicating whether an input instruction is a member of a set of instructions. The address generator outputs an address based on the input instruction. A jump instruction generator outputs an instruction to jump to the address. The system also includes an instruction selector that has inputs in communication with the

source of instructions, the jump instruction generator and the complex instruction detector. Here, the instruction selector outputs either the instruction from the source or the jump based on the complex instruction detector output.

Independent claims 26 and 37 also require a complex instruction detector, an address generator, a jump instruction generator, and an instruction selector. In claim 26, the instruction selector is "connected to the jump instruction generator, the memory, the complex instruction detector and the processor so as to receive: jump instructions from the jump instruction generator; computer instructions from the memory; and the value from the complex instruction detector, whereby depending on the value from the complex instruction detector, either the jump instruction or the computer instruction is provided by the instruction selector to the processor." In claim 37, the instruction selector is "operatively connected to the jump instruction generator, the memory, the complex instruction detector, and the processor so as to receive jump instructions from the jump instruction generator, computer instructions from the memory, and the value from the complex instruction detector; whereby depending on the value from the complex instruction detector, either the jump instruction or the computer instruction is selected by the instruction selector and is provided by the instruction selector to the processor."

Claim 36 is a method of executing a program, and includes the steps of "generating a jump and link instruction to the address, the jump and link instruction comprising an instruction for the processor to execute instructions at the address and then return to the instruction following the original instruction in the program; determining whether the original instruction is a complex instruction on an instruction by instruction basis; and selecting between the jump and link

instruction and the original instruction based on the result of the determining."

As discussed above and illustrated in the accompanying Appendix, *Blomgren* requires a serial process having two passes through the system and two attempts by a processor (execution unit) to execute a complex instruction. If the first attempt fails, a new, emulated version of the instruction is passed through the system. *Blomgren's* multiplexer does not select between two versions of the instruction as in claim 36, or output either a source/computer instruction or jump instruction as in claims 1, 26 and 37, because only one version of the instruction is placed in the instruction fetch at a time - either the original instruction in pass A or the emulated instruction in pass B. In contrast, the claimed invention efficiently outputs only one of two versions of an instruction via the instruction selector.

Therefore, for at least these reasons, *Blomgren* lacks a disclosure or teaching of all of the elements of independent claims 1, 26, 36 and 37. Thus, applicant respectfully requests reconsideration and allowance of these claims. Claims 2, 4-5, 9-11 and 27-30 depend from claims 1 and 26, respectively, and contain all of the limitations thereof as well as other limitations that are neither disclosed nor suggested by the prior art of record. Accordingly, applicant submits that the dependent claims are likewise patentable.

Claims 15-22, 31, and 33-34 were rejected under 35 U.S.C. § 103(a) as being obvious over *Blomgren* in view *Hennessy*. Applicant respectfully traverses the rejection. The Examiner suggested several modifications to claim 31 in order to overcome the rejection. In view of the suggested modifications, claim 31 has been amended to recite "an address generator connected to the source and a jump instruction generator, the address

generator receiving computer instructions from the source and, if a received instruction is complex, providing an address in a memory containing emulation instructions to the jump instruction generator, whereby the emulation instructions are simple instructions and emulate the intended function of the complex instruction" as well as "wherein the complex instruction detector is configured to operate on the received instruction in parallel with the address generator and the jump instruction generator operating on the received instruction."

Independent claims 23 and 35 have also been amended in view of the Examiner's suggestions pertaining to claim 31. Specifically, claim 23 has been amended to recite "wherein the step of determining whether the original instruction is a member of the set of instructions is performed in parallel with generating the address and generating the jump and link instruction." Claim 35 has been amended to recite "a jump instruction generator in operative communication with the address generator, the jump instruction generator being operable to receive information from the address generator and to output an instruction to jump to the address output from the address generator based upon the information" and "wherein the complex instruction detector is configured to operate on the computer instructions in parallel with the address generator operating on the computer instructions and the jump instruction generator operating on the information received from the address generator."

Amended claim 15 recites, among other limitations, "providing both the computer instruction and the jump instruction; and selecting the jump instruction or the computer instruction based on the result of the step of determining; wherein determining whether the computer instruction is complex is performed in parallel with the steps of generating the

address and generating the jump instruction." As discussed above with regard to the anticipation rejection of independent claims 1, 26, 36 and 37, *Blomgren* functions in a serial two pass process. In the first pass, *Blomgren's* processor initially attempts to process an instruction, and if it cannot, then the second pass obtains another version of the instruction that the processor is capable of handling. In contrast, the determining step of claim 15 is performed in parallel with the address and jump instruction generation steps, both the computer instruction and the jump instruction are provided, and one or the other is selected.

Thus, it is respectfully submitted that amended independent claims 15, 23, 31 and 35 patentably distinguish over *Blomgren* and *Hennessy*, both individually and in combination, as the references do not teach or suggest performing the claimed operations in parallel in the manners indicated above. Therefore, applicant respectfully requests reconsideration and allowance of these claims.

Claims 16-22 and 33-34 depend from claims 15 and 31, respectively, and contain all of the limitations thereof as well as other limitations that are neither disclosed nor suggested by the prior art of record. Accordingly, applicant submits that the dependent claims are likewise patentable.

Claims 6-8 and 12-14 were rejected under 35 U.S.C. § 103(a) as being obvious in view of *Blomgren*. Claims 3 and 24-25 were rejected under 35 U.S.C. § 103(a) as being obvious over *Blomgren* in view of U.S. Patent No. 5,826,089 to *Ireton* ("*Ireton*"). Claim 32 was rejected as being obvious over *Blomgren* in view of *Ireton* and *Hennessy*. For at least the reasons discussed above, independent claims 1, 15, 23, 26, 31, and 35-37 are patentable. Claims 3, 6-8, 12-14, 24-25 and 32 depend from independent claims 1, 23 and 31, respectively, and

contain all of the limitations thereof as well as other limitations that are neither disclosed nor suggested by the prior art of record. Accordingly, applicant submits that the dependent claims are likewise patentable.

As it is believed that all of the rejections set forth in the Office Action have been fully met, favorable reconsideration and allowance are earnestly solicited. If, however, for any reason the Examiner does not believe that such action can be taken at this time, it is respectfully requested that she telephone applicant's attorney at (908) 654-5000 in order to overcome any additional objections which she might have.

If there are any additional charges in connection with this requested amendment, the Examiner is authorized to charge Deposit Account No. 12-1095 therefor.

Dated: October 26, 2005

Respectfully submitted,

By 

Andrew T. Zidel
Registration No.: 45,256
LERNER, DAVID, LITTENBERG,
KRUMHOLZ & MENTLIK, LLP
600 South Avenue West
Westfield, New Jersey 07090
(908) 654-5000
Attorney for Applicant

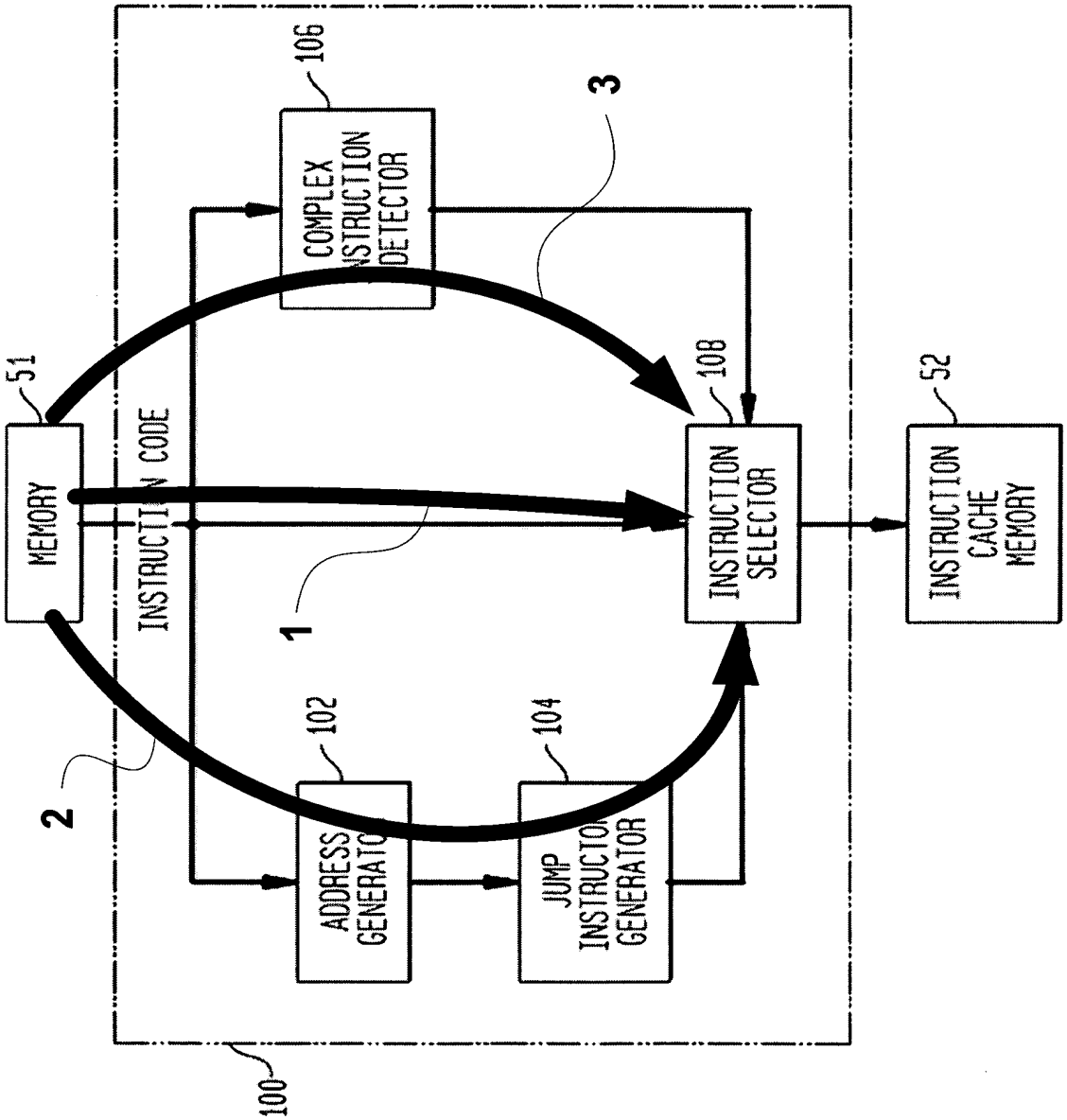
599596_1.DOC

Appendix A

FIG. 2 of Application

FIG. 2 of *Blomgren*, U.S. Pat. No. 5,781,750

FIG. 2



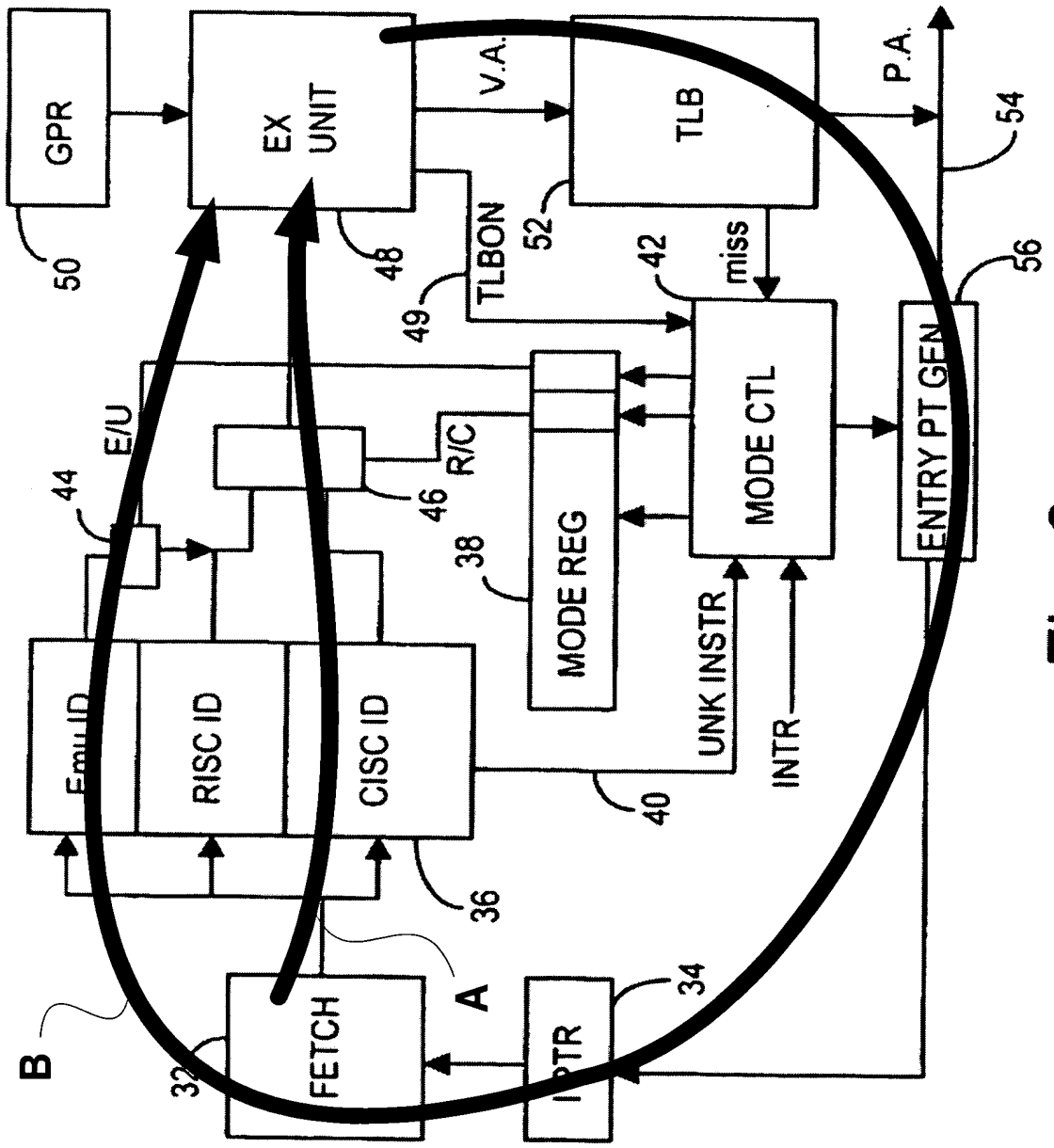


Fig. 2